Hospital Management Knowledge Discovery using Discrete Event Simulation

Christopher A. Bain
Management Information Systems and Modelling Group, CEHSEU
Melbourne Health
Grattan St, Parkville 3050, Victoria
Christopher.Bain@mh.org.au

Gitesh K. Raikundalia
School of Computer Science and Mathematics
Victoria University
PO Box 14428 Melbourne City 8001, Victoria
Gitesh.Raikundalia@vu.edu.au

Snehal Mehta
Management Information Systems and Modelling Group, CEHSEU
Melbourne Health
Grattan St, Parkville 3050, Victoria
Snehal.Mehta@mh.org.au

Abstract

This paper describes research into one of the key areas of knowledge in healthcare. That is, the knowledge base of hospital managers in relation to access, demand and resource management in hospitals. The latter are ongoing problems which impact on patients in the form of long emergency waits, extended periods on procedural waiting lists and cancelled surgical procedures. Hospital managers have the unenviable task of contributing to and maintaining knowledge bases in a number of dimensions, including both the clinical and managerial dimensions. Whilst there are evolving knowledge bases in the clinical domain, for example, expressed in protocols, guidelines and decision support rules, there is nothing equivalent in the hospital management domain. There is certainly no work that integrates both perspectives. This paper outlines a framework and mechanism within which knowledge bases for healthcare managers can be established and utilised.

Keywords: Hospital knowledge management simulation, discrete event simulation.

1 Introduction

The research problem dealt with in this work is the question of how to develop, maintain and utilise knowledge bases for healthcare managers, particularly in relation to the access, demand and resourcing problems that these managers face on a daily basis.

In addition, hospitals have no agreed "truths" in relation to these issues and no corporate memory in relation to knowledge and experience in these areas. Hospitals certainly have no means to integrate the clinical and managerial perspectives regarding these issues.

In particular, this paper describes a software framework in which user-defined simulation models can be created and run via a Web browser. Hospital managers can explore solutions to these problems, and hence build a knowledge base that can be called upon subsequently, in an electronic form, to guide future decision-making.

Over the years, simulation applications (Harper 2002, Bowers and Mould 2005) have been created using various technologies and platforms. Specific applications have been developed based upon these software tools to create and run simulations in order to obtain the required results. The applications are developed to work on a single machine, multiple machines, over an intranet or over the Internet.

Our work involves creation and run of simulations using a discrete event simulation engine over an intranet or the Internet. The paper discusses a Web-based application that has been developed using these technologies to create simulation objects with required parameters and return results of simulation. Since this application requires very limited resources, it can be implemented by any organisation that has multiple users creating and running different simulation models.

In summary, we have developed an end-to-end (user to simulation, and back to user) simulation framework in which a range of user-defined simulation models can be run. The direct implications of this work are that:

- Users can define the models they need to suit the business problem they are analysing.
- Standardised interfaces can be developed to allow users familiarity with the process of defining a simulation model to represent a given business problem, independent of the business problem being analysed.
- The work will facilitate the development of standard languages and communication media (e.g., XML standards) to capture the business knowledge in this domain, and facilitate the subsequent development of even better technologies in this area.

The importance of this work for the healthcare industry is that:

- Hospital managers can begin to think about and define their access, demand and resourcing problems in ways that can be directly solved by these kinds of technologies.
- Integration across clinical and management decision-making can be achieved. For example, document if there is an influx of influenza patients into a hospital, as often happens each winter. In this example, we know that there will be an influx of influenza patients into hospitals and we also know this will result in more cases of pneumonia and also some deaths. We have no means, however, of modelling the combined clinical (cases of pneumonia and deaths) and managerial (e.g., extra numbers of beds required, more nurses required to staff beds) impacts of this phenomenon.
- The cost of using these technologies in the already "cash-strapped" public healthcare sector can be minimised.

2 Literature Review

There is a growing body of literature regarding the application of simulation modeling in the healthcare setting. Pitt (Pitt 1997) reports a simulation modelling system that can support the development of multiple simulation models, directly accessible by managers. The key weakness of that work, in relation to knowledge discovery, is that users need to install separate instances of the application across multiple machines that are not capable of communicating with each other. In the system we have developed however, users can view the same model or variations of it across the entire organisation (or even over the Internet) such that the knowledge gained can be readily agreed upon, shared and actioned.

There are many examples of simulation applications that have been developed to address distinct hospital departments (e.g., Emergency Departments) in great detail (Sinreich and Marmor 2004, Evans et al 1996, Blasak et al 2003, Hannan et al 1974). Similar work has been done in relation to Operating Theatres (Ferrin et al 2004) and Outpatient Departments (Wijewickrama and Takakuwa 2005). There are even applications for pharmacy operations (Spry and Lawley 2005). Our work distinguishes itself in this regard by being able to examine hospitals from a micro-level all the way up to a macro-level, across an entire hospital or in one specific department or sub-process, although additional GUI development work is required to improve this functionality further.

There have been some attempts to create simulation models that can be used, with little modification, in a range of different hospitals. For example, Lowery's work (Lowery 1992, Lowery 1993) on building a critical care model. What this and other work (Lowery, 1996) do not do however, is address how simulation software can be integrated into the broader hospital IT environment (e.g., on the Web, using standardised data files – via XML in our case) so as to provide maximal use to end-users and other stakeholders. Nor does such work address how such an application can then be used in the area of knowledge discovery and knowledge capture for an organisation.

Other authors (Schenk et al 2005) have performed work on the analytical and mathematical aspects of simulation. Our work can be distinguished by being able to draw on more technical or theoretical work in order to translate those to business knowledge and business value for healthcare managers.

3 Historical Context

There are many examples in the literature where the business problems confronting healthcare managers are quoted, and rightly so, as the justification for the background research that various forms of management decision support, including simulation modelling (Rosko 1999, Bowers and Mould 2005). Yet there remains a huge gap between the research and practical solutions to these problems. We suggest that one reason is the lack of standards in this area so that evolving technologies can be developed in accordance or congruence with these standards. Another reason is that applications that are being developed are perhaps not being developed using technologies that suit the context in which they are placed. For instance, it is no use to a health manager to build a solution in an obscure mathematical programming language.

A range of research and development activities in the areas of modelling and simulation are being undertaken in the relevant technical communities. These include, but are not limited to, information technology and computer science, mathematics and statistics, operations research, health economics and epidemiology. However, there remains a gap in terms of a common taxonomy or understanding of modelling frameworks or the best ways in which to describe relevant business problems such that different technical solutions can be applied to a common description of the same problem.

Although there has been some preliminary work in this area, there are no clearly agreed standards in healthcare in relation to naming or describing predictive technologies – including models of various kinds.

In order to advance the background science in this area to the point of practical application, there is a necessary role for standards (McCabe 2005) to harness existing activity from around the world so that problems defined in a standard way can have a range of technologies applied to them. In turn, stakeholders can then best evaluate and choose between proposed solutions to their business problems as well as having choice regarding cost and suitability of technical solutions to their current environment.

As alluded to previously, knowledge bases in the clinical domain are growing in number and complexity across a range of clinical scenarios. From decision support guidelines expressed in clinical applications (Tsumoto et al 2003) to protocols in chemotherapy provision in oncology (Bury et al 2004) and so forth. In fact, not only are those knowledge bases growing and increasing in complexity, in many cases they are already incorporated into computer systems that are used in routine clinical care. In terms of knowledge bases and knowledge discovery in healthcare, many of the biggest gaps are really in the management domain, particularly regarding integrating clinical knowledge with management. Consider the previous example of influenza. In most hospitals there is no corporate knowledge that healthcare managers can draw on to answer questions such as, "What does a typical winter mean for a hospital under a range of loads of influenza?". We know influenza appears every year, but we do not capture the corporate knowledge, particularly from the managerial perspective, that states something like:

- Mild flu outbreak: require 7 extra beds
- Moderate flu outbreak: require 14 extra beds
- Severe flu outbreak: require 21 extra beds.

Taking it to the next level, there is even less available about corporate knowledge or agreed truths regarding what the effect of an outbreak as described above will be, if we also factor in the possibility of bed closures. These closures occur due to staff sick leave since some staff is also likely to be affected by the flu. Models developed in our system can be used to examine the effect of increased general and special (e.g., Intensive Care Unit) bed use, staff sick leave and the follow-on budgetary and access consequences under a range of scenarios like these.

In turn, the system allows a hospital to establish a range of clear lessons, or at least principles, about how such an epidemic would affect their institution, and what mitigating actions could be taken to manage the adverse effects on the organisation. In other words, the system can allow an entire body of knowledge to be discovered and archived in preparation for these real-world scenarios.

We will refer to the influenza example throughout this paper as a means of explaining how the system described can assist in the broader aim of establishing knowledge bases for hospital managers and hospitals as organisations.

Technologies can be used to answer key access, demand and resourcing questions. For instance, the question of "How many beds will we need to set side for elective vascular surgical activity in a year's time?". Each question will imply a different type and level of solution from a technical perspective. However, this framework and approach allows a means of standardisation across a range of problems like these.

Stakeholder engagement is also an important issue in this problem domain (Garland et al 2004), especially on the healthcare administration and management side of the equation (Bain and Au 2006). Medical staff engagement is also seen as an issue in obtaining system change in hospitals and health services (Gollop 2004). This framework and approach highly involves managers and senior clinicians by bringing standard interfaces to them and allowing them to engage with such systems in a familiar Web-based paradigm.

4 Web-based simulation application

The system we have developed requires very limited resources to operate. The system requires a computer with a Web browser installed and connected to the Internet or an intranet. The server will require a Web server and *Simlu8* software (SIMUL8 Corporation 2006) to create and run simulations. This system uses Simul8 simulation software to create and run discrete event simulation models. Users interact with the system using a very simple browser-based interface. Users can create a model by following some steps to create model objects with their parameters.

For example, in the context of a flu epidemic, as mentioned earlier, this can be very useful. A user could build a model, or several models, which allows the user to examine the impact of the epidemic from both a clinical and managerial perspective. On the clinical side, users (such as senior doctors) can determine the number of cases of pneumonia that will be likely to present to the hospital, based on data they have about the catchment area and the rate of conversion of flu cases to pneumonia cases. On the managerial side, these same users or other users such as hospital managers can determine the likely bed requirements based on the scenario in Section 2 above and the cost in terms of nursing resources, investigations and treatment. Previously, such scenarios could usually only be explored with localised simulation models, if indeed managers were fortunate enough to have one available and to be trained in its use.

The ability to use this application and framework will allow managers to build up their knowledge bases regarding appropriate management strategies in contexts such as a flu epidemic or other such contexts. Once a model's skeleton is created, the user needs to start the simulation model and the system will automatically generate a model in Simlu8. Users will not be able to view any of this processing that will occur in the background on the Web server. Users only need to wait for their results to be ready. Once the simulation is completed, users will be able to view their results on their Web browser.

4.1 Use of Simul8

Simul8 © (Version 12.0) is simulation software that is used to create and run simulation models. One can create a simulation model by creating individual objects (for instance, in our case, work centres have a natural correspondence to beds in a hospital) or processes and linking them. Each process or object can have constraints or parameters associated with them to control their operation. Again, in the case of a hospital bed (work centre), there are resources such as nursing staff, and these resources can have a cost attached to them in a model, just as in the real world. One can view the actual working of a model when the simulation model is executed. It shows each processing element or item passing through various stages simulating the operation in an actual working environment.

We can apply an XML document to Simul8 to create models and set parameters. The XML document has specific predefined elements for creating and connecting different objects to run a simulation.

4.2 Architecture of the application

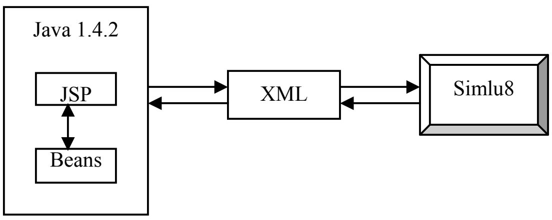

Figure 1: Sub-architecture

The application uses Java as the main technology to create models as well as extract and display results. Figure 1 shows the architecture involving Java, XML and Simul8.

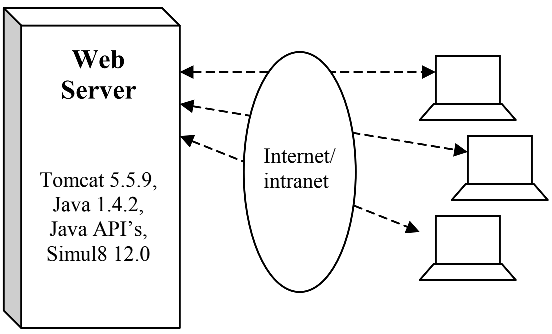

Figure 2: System architecture

The basic architecture of the system is shown in Figure 2. To run this application, a Web server is required with Tomcat installed. There is no requirement of any database to store the models created. The models are stored in a user-created file that stores the object descriptions.

5 System Functionality

Melbourne Health is the health research organisation involved in this project. Currently, the application is implemented on the Melbourne Health intranet. The intranet uses a Web server to host the application that can be accessed by anyone from Melbourne Health.

To understand the operation of the system, it is important to understand the basic architecture of the system. As shown in Figure 1, Simlu8 can interpret XML documents to create and run a model.

The key functionality of the system can be defined as follows:

- Users are allowed to define their access, demand and resourcing problems in a standard way.
- Models are run to explore scenarios related to these problems.
- Results of simulation are returned to the user for comparison across scenarios.

This has significant implications for hospital managerial knowledge discovery and paves the way for future work (see later section).

To run this system, the user needs to enter a correct URL in an Internet browser pointing to the application source location. The user is asked to register if not previously registered.

Once a user is registered, a directory is created for that particular user that can hold all the models created by the user. The user can log into the system using the username and password selected at the time of registration.

User login is authorised and a project management page is displayed as shown in Figure 3. The user is able to create new projects, open existing projects and view results of previously executed projects using this page.

Different projects can hold different simulation models. To create a new project, the name of the project is entered in the text field provided. This refreshes the page and a new project name will be available in the drop down menu list. This menu list holds all the projects created by the user. In the context of a hospital, different projects may be, for example, a model that analyses the flu epidemic scenario described above, or a completely different scenario such as an examination of the effect of closing down some surgical services over the Christmas period. Alternatively, a range of projects may explore a number of subtle variations of the core scenario of a flu epidemic. For example, the scenario under a range of different extents of the epidemic, from 10% of the potential at-risk population being affected by flu to 45% of the potential at risk population being affected by flu.

The results of models that have already been executed can also be viewed. The names of projects under "List of your Projects" are hyperlinks displaying the results of the various projects.

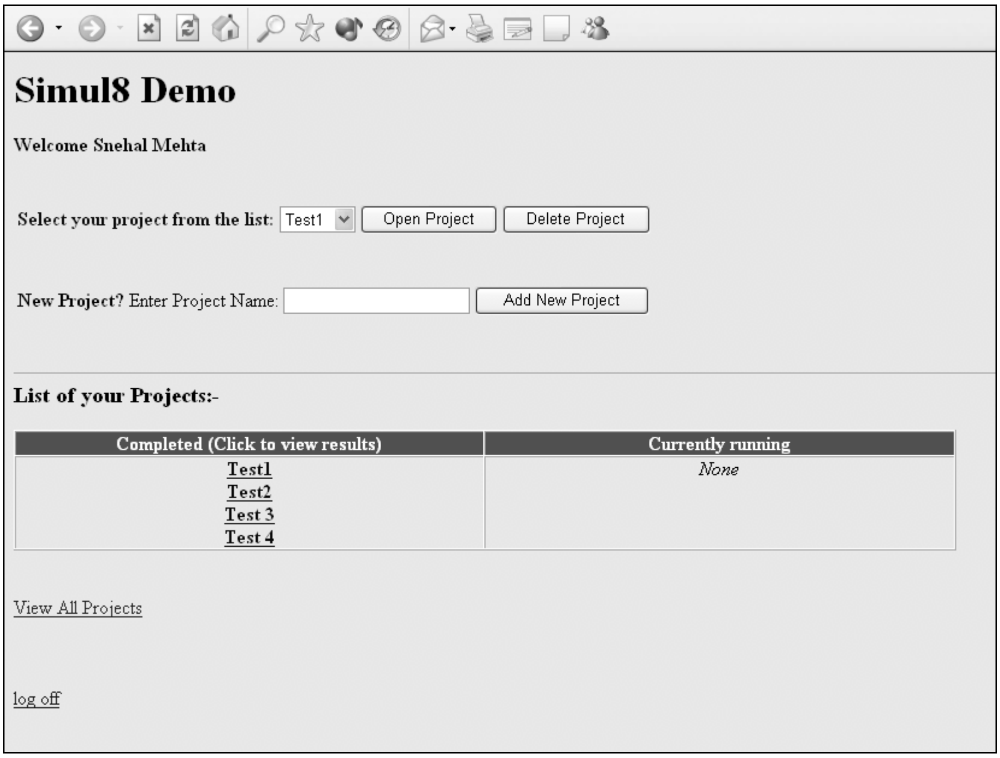

Figure 3: Project Management page

Projects created by all users can be listed using the 'View All Projects' hyperlink. This will display the list of all completed projects and projects currently run by all users of the system. However, the *results* of these completed projects of other users cannot be viewed.

Once a new project is open, the user is shown a page (Figure 4) to create model objects or edit previously created objects. To create a model, the user has to start with creating work entry points (WEs) that act as the entry points of the model. Each object of the model receives a unique ID that is automatically generated by the system.

When a WE is created by entering the name of it (as shown at the top of Figure 4), a WE object is created with a default parameter value. The parameter associated with the WE is the inter-arrival time of the items entering the system. This time must have a value and type of distribution. In the context of a hospital, a WE may, for example, be the "feeder stream" or arrival point into the hospital of a particular group of patients (for instance, flu patients coming into emergency). Multiple WEs may exist in any given model. For example, a complex model of the flu scenario may have arrival streams (via WEs) of:

- flu patients arriving via emergency but destined for discharge home
- flu patients arriving via emergency but destined for admission, and
- flu patients arriving directly to the Intensive Care Unit because they have already advanced to the stage of severe pneumonia.

The user can edit the parameter value and distribution type by using 'Edit Parameters'. The user can modify the parameter's type and the inter-arrival time. This can be used to represent the pattern and volume of arrivals of each group of patients.

'Edit' and 'Delete' allow modification and deletion of the WE object, respectively. The user can create more than one WE by entering the names of new WE objects and pressing 'Add'.

Next, objects can be created as storage queues. A storage queue can act as a reception or waiting area of a hospital. In the case of our flu example, objects may represent the queue of patients waiting to be seen in emergency upon arrival. These objects are created in a similar way to WE objects, but the user needs to select a parent WE name for each queue object. A link is created between the parent WE object and the queue object. A storage queue can have more than one parent object selected. One can specify the capacity of the queue and the wait time while creating these objects. The default is unlimited queue capacity with no waiting time.

After creating queue objects, work centre (WC) objects are created. These are the objects where the actual work takes place.

To create a WC object, the WC name and number of instances for that object are entered. For example, if we need 10 wards in a hospital model, the name of the WC is entered as 'Ward' and '10' is entered as the number of instances. Parent queue objects from where the work items are connected are selected. Default timing parameters are associated with the WC objects when the object is added. Later these parameters can be edited by

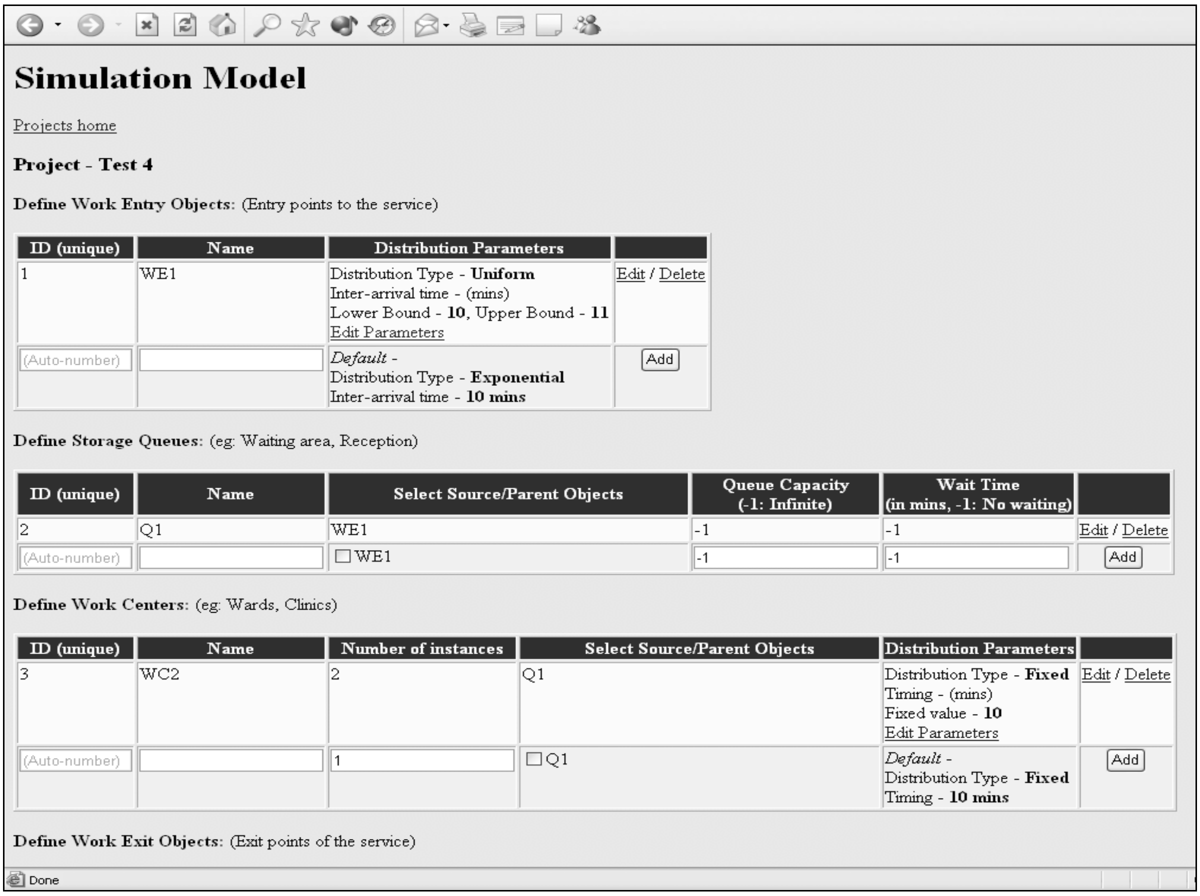

Figure 4: Model Creation page

clicking 'Edit Parameters' (Figure 5). In our flu example, the timing parameters can be used to describe the amount of treatment and recovery time in a hospital bed for a given type of flu case, dependant upon their location. For example, the time taken for such patients in an ICU can follow a particular distribution, which the model can represent. In addition, the resources such as the nursing and investigation resources attached to that period of stay in that bed type can be represented in very specific ways, and the related costs are calculated. A very different set of timing parameters and resource data can be attached to a standard ward bed, as in the real world.

In this very simple example, it can be seen that large amounts of valuable information can be generated to assist hospital managers in improving corporate knowledge bases regarding their problem domain, in a controlled and readily accessible environment, in a way that cannot be easily done in the real world.

After creation of WC objects, work exit point (WEX) objects have to be created. The section of the page for this is not shown at the bottom of Figure 4, but is similar to the above sections of the same page. These objects act as the exit point from the system. WEXs do not have any parameters associated with them but we need to select a parent WC as their input links while adding each WEX object. The real world equivalent of a WEX is the discharge or transfer of a patient to another institution.

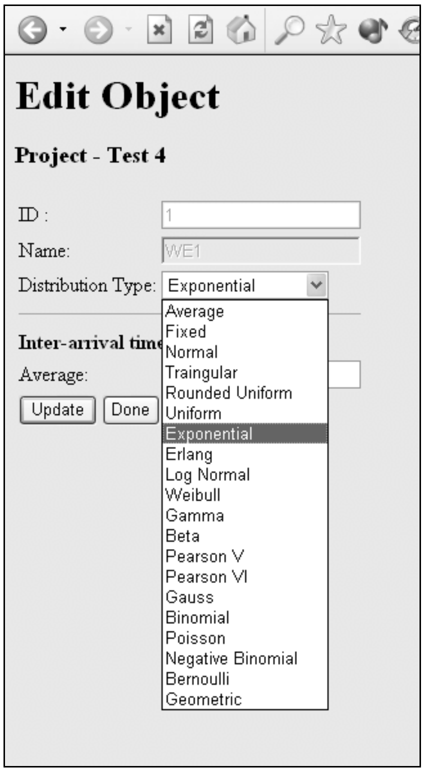

Figure 5: Editing parameters

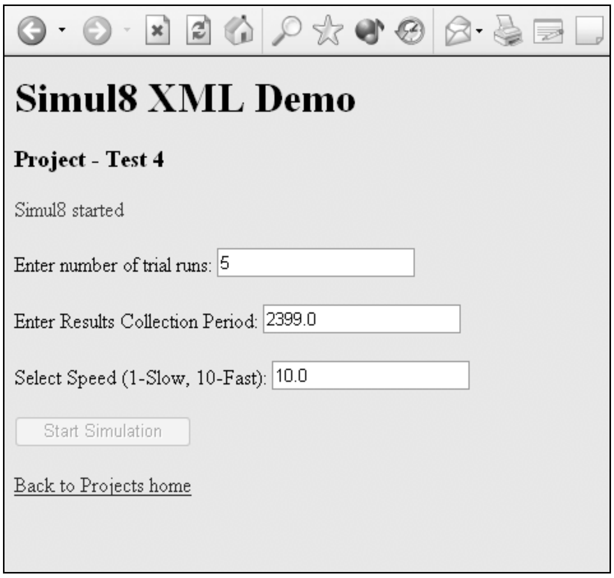

Figure 6: Start simulation

After this information is entered, the 'Start Simul8 Model' link at the bottom of the model creation page (not shown) is taken. This will open a page to enter simulation model parameters before running the actual simulation model.

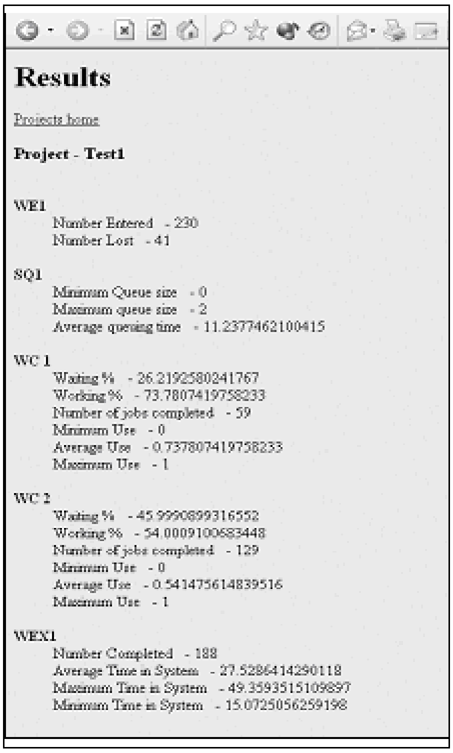

Figure 7: Results page

Figure 6 shows the parameters required to run a simulation model. Once these parameters are updated, the user needs to click 'Start Simulation' to run the simulation and view the results. The 'number of trial runs' allows the user to determine how many model trials are run. Given that many of the input parameters are distributions (as mentioned previously), this critical input determines how many trials are run so that a "typical" or representative result is obtained. The 'Results Collection Period' basically allows the user to describe how long (in virtual time) he/she wishes to run the simulation for. In our flu example, the user may wish to gain knowledge about the "behaviour" and effect of the flu influx for a month, over the entire winter period, or over each winter for the next 3 years.

After this point, the user will be unable to view the actual simulation being executed as it is executed on the Web server.

To view the results of the simulation, the user needs to go back to the project management page (Figure 3). If the simulation model is still being executed on the server, the name of the project will appear in the currently running projects list. The user needs to refresh this page to check if the simulation model has been completed and generated the results[1]. When the project name appears under the completed projects list, the user can view standard simulation results as shown in Figure 7.

An example of some dynamic XML generated through model creation is shown in Appendix A1. Part of an XML results document is shown in Appendix A2.

6 Case Study

Let us more closely examine the example used previously regarding an influenza outbreak. This case study demonstrates how a user would interact with the system to establish a knowledge base regarding the problem.

First, the user creates a new project via the project management page. For example, "2006-78 Winter Flu Project-Base Scenario". The user would then navigate to the model creation page and would enter 2 WEs: one WE representing the standard patient load into the hospital, the other WE representing the increase in load from flu patients needing admission.

The user then would enter a hospital size, for instance, 350 beds (350 WCs), and the relevant WE and storage area parameters, which connect the key components of the model (WEs and WCs) and allow it to function.

After navigating to the start Simul8 Model page, the user can enter the run parameters, such as the number of trials, and commence the simulation. The user can then navigate to the results page to examine the results of the project.

In order to build a knowledge base regarding the problem, the user can repeat the above steps and compare results across multiple projects. The projects may, for example, each differ by minor changes in flu load conditions, or numbers of beds available, so that queues, bottlenecks and resource implications can be examined.

7 Discussion

The above work describes the current state of development of an ongoing program of work aiming to

[1] Of course a technical solution can be created, and will be pursued as further work, which automatically updates the page with information without requiring a refresh.

deliver a suite of software tools and decision support aids to hospital managers to assist them in their daily work. Given that many of the problems reported by patients in relation to their interactions with the hospital system regard access to care and inconvenience, this work is in turn intended to provide value to our patients.

Future work that can follow on from this is:

- Communication between simulation models. For instance, between two interconnected facilities within one health service or across a city or anywhere in between. The practical application can be, for example, in load balancing for elective and emergency mix in Winter, or for the best site to apply a large funding boost for elective surgery.
- Communication between live systems and simulation models.
- A common means for describing business scenarios to be simulated, irrespective of the technology that is being used as a simulation engine.
- Potential standardisation of decision interventions that can be trialled in a virtual environment, which relates to real-life decision-making scenarios (again, refer to the first point about communication between live systems and simulation models).
- Further knowledge discovery for healthcare managers in this domain. This has implications operationally and for training new hospital managers.

8 Conclusion

This paper demonstrates the importance of a simulation-based Web framework for exploring complex hospital management problems across both clinical and managerial dimensions. This system will thereby enable the development of key hospital management knowledge bases, which until now have been difficult to define and store. Furthermore, the system will facilitate further development in this domain, such as simulation models that can communicate between each other and with real-time operational systems.

9 References

Bain, C. and Au, L. (2006): A framework for using predictive technologies in healthcare management. Submitted to *Aus Health Rev.* July 2006.

Gollop, R., Whitby, E., Buchanan, D. and Ketley, D. (2004): Influencing sceptical staff to become supporters of service improvement: a qualitative study of doctors' and managers' views. *Qual Saf Health Care* **13**:108-114.

Harper, P. (2002): A framework for operational modelling of hospital resources. *Healthcare Management Science* **5**:167-173.

Bowers, J. and Mould, G. (2005): Ambulatory care and orthopaedic capacity planning. *Healthcare Management Science* **8**:41-47.

SIMUL8 Corporation (2006): SIMUL8, SIMUL8 Corporation, http://www.simul8.com/. Accessed 29 Sept 2006.

Rosko, M. (1999): Impact of internal and external environmental pressures on hospital inefficiency. Healthcare Management Science. 1999;2(2):63-74.

Garland, A., Lewczyk-Boxmeyer, C., Gabayan, E. and Hawley, K. (2004): Multiple stakeholder agreement on desired outcomes for adolescents' mental health services. *Psychiatric Services* **55**:671-676.

McCabe, K. (2005): IEEE begins to revise four simulation standards, http://standards.ieee.org/announcements/pr_simulation.html. Accessed 2 Oct 2006.

Tsumoto, S., Hirano, S. and Hanada, E. (2003): Internet-based decision support: towards e-Hospital. *Proc. 27th Annual International Computer Software and Applications Conference*. http://csdl2.computer.org/persagen/DLAbsToc.jsp?resourcePath=/dl/proceedings/&toc=comp/proceedings/compsac/2003/2020/00/2020toc.xml&DOI=10.1109/CMPSAC.2003.1245401. Accessed 2 Oct 2006.

Bury, J., Hurt, C., Roy, A., Bradburn, M., Cross, S., Fox, J. and Saha, V. (2004): A quantitative and qualitative evaluation of LISA, a decision support system for chemotherapy dosing in childhood acute lymphoblastic leukaemia. Medinfo. 2004; 11(Pt 1): 197-201.

Pitt, M. (1997): A generalised simulation system to support strategic resource planning in healthcare. *Proc. 1997 Winter Simulation Conference*, Atlanta, USA, 1155–1162.

Sinreich, D. and Marmor, Y. (2004): A simple and intuitive simulation tool for analysing Emergency Department Operations. *Proc. 2004 Winter Simulation Conference*, Washington DC, USA, **2**:1994–2002.

Evans, G., Gor, T. and Unger, E. (1996): A simulation model for evaluating personnel schedules in a hospital emergency department. *Proc. 1996 Winter Simulation Conference*, Coronado, USA, 1205-1209.

Blasak, R., Starks, D., Armel, W. and Hayduk, M. (2003): Healthcare process analysis: the use of simulation to evaluate hospital operations between the emergency department and a medical telemetry unit. *Proc. 2003 Winter Simulation Conference*, New Orleans, 1887-1893.

Ferrin, D., Miller, M., Wininger, S. and Neuendorf, M. (2004): Analyzing incentives and scheduling in a major metropolitan hospital operating room through simulation. *Proc. 2004 Winter Simulation Conference*, Washington DC, USA, **2**:1975-1980.

Hannan, E., Giglio, R. and Sadowski, R. (1974): A simulation analysis of a hospital emergency department. *Proc. 1974 Winter Simulation Conference*, Washington DC, USA, 379 – 388.

Spry, C. and Lawley, M. (2005): Evaluating hospital pharmacy staffing and work scheduling using simulation. *Proc. 2005 Winter Simulation Conference*, Orlando, USA, 2256-2263.

Wijewickrama, A. and Takakuwa, S. (2005): Simulation analysis of appointment scheduling in an outpatient department of internal medicine. *Proc. 2005 Winter Simulation Conference*, Orlando, USA, 2264-2273.

Lowery, J. (1992): Simulation of a hospital's surgical suite and critical care area. *Proc. 1992 Winter Simulation Conference*, Arlington, USA, 1071-1078.

Lowery, J. (1993): Multi-hospital validation of critical care simulation model. *Proc. 1993 Winter Simulation Conference*, Los Angeles, USA, 1207-1215.

Lowery, J. (1996): Design of hospital admissions scheduling system using simulation. *Proc. 1996 Winter Simulation Conference*, Coronado, USA, 1199-1204.

APPENDIX

A1


```
<SIMUL8XML>
    <SimulationParameters>
        <Trial>
            <Runs>5</Runs>
        </Trial>
        <ResultsCollectionPeriod>2399.0</ResultsCollectionPeriod>
        <ResultsSummary>
            <ResultItem>
                <ObjectID>1</ObjectID>
                <Code1>1</Code1>
                <Description>Number Entered</Description>
            </ResultItem>
          |          |          |          |
        <SimulationObject Name='WEX1' Type='Work Complete' ID='13'>
            <DisplayData>
                <X1>500</X1>
                <Y1>75</Y1>
                <X2>512</X2>
                <Y2>87</Y2>
            </DisplayData>
          |          |          |          |
        </SimulationObject>
          |          |          |          |
</SIMUL8XML>
```


A2


```
<?xml version="1.0" encoding="UTF-8" ?>
<SIMUL8XML>
  <SimulationStatus>
    <SimulationTime>2399</SimulationTime>
    <ResultsCollectionPeriod>2399</ResultsCollectionPeriod>
    <Warmuptime>0</Warmuptime>
    <S8FileName>C:\Projects\Simul8XML\snehal\Test1\MySimfile.XML</S8FileName>
    <Trial>
      <Title></Title>
      <Notes></Notes>
      <Runs>5</Runs>
      <ConfLimitPercent>95.0</ConfLimitPercent>
      <TrialResultsValid>No</TrialResultsValid>
    </Trial>
    <RunComplete>Yes</RunComplete>
  </SimulationStatus>
  <Results>
    <ResultsSummary>
      <Result Position="1" ID="1" Code1="1" Code2="0">
        <ObjectName>WE1</ObjectName>
        <Description>Number Entered</Description>
        <RunResult RunNumber="0">
          <Value>230</Value>
          <RNStream>1</RNStream>
        </RunResult>
      </Result>
        |          |          |          |
```